NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

SEARCH FOR A STEALTHY FLIGHT PATH THROUGH
A HOSTILE RADAR DEFENSE NETWORK

by

John J. Leary III

March 1995

Thesis Advisors: R. Kevin Wood
Craig W. Rasmussen

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 1995	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE SEARCH FOR A STEALTHY FLIGHT PATH THROUGH A HOSTILE RADAR DEFENSE NETWORK	5. FUNDING NUMBERS
6. AUTHOR(S) John J. Leary III	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER Operations Research Dept. Naval Postgraduate School
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. Approved for public release; distribution is unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

This thesis develops a method for quickly selecting a good flight path for an aircraft flying from its base to its mission objective when the flight path must be over a geographic area containing hostile radar installations. Several models are developed in detail and then integrated permitting a shortest path algorithm to be used to find a route from a starting location to a goal that approximately minimizes probability of detection. Fuel, time and distance constraints are incorporated indirectly, but one strong assumption is made: Detections are assumed independent across network edges and among different radars. A test program is written in FORTRAN and run on a desktop PC using a battery of tests to validate the model. Problems designed with predictable paths having zero probability of detection are solved in a few hundred or thousand seconds for the predicted, optimal paths. Results on problems having non-zero probabilities of detection are less conclusive, but indicate that the method has promise. Errors due to computations and assumptions, as well as their bounds, are discussed, as are recommendations for further model development.

14. SUBJECT TERMS Flight Planning, Stealth, Radar, Network Search, Routing, Shortest Path, Probability of Detection	15. NUMBER OF PAGES 85
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

Approved for public release; distribution is unlimited.

SEARCH FOR A STEALTHY FLIGHT PATH THROUGH A HOSTILE RADAR DEFENSE NETWORK

John J. Leary III
Lieutenant, United States Navy
B.S., United States Naval Academy,1983

Submitted in partial fulfillment
of the requirements for the degrees of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1995

Author: John J. Leary III

Approved by: R. Kevin Wood, Thesis Advisor

Craig W. Rasmussen, Thesis Advisor

Gerald G. Brown, Second Reader

Richard Franke, Chairman
Department of Mathematics

Peter Purdue, Chairman
Department of Operations Research

ABSTRACT

This thesis develops a method for quickly selecting a good flight path for an aircraft flying from its base to its mission objective when the flight path must be over a geographic area containing hostile radar installations. Several models are developed in detail and then integrated, permitting a shortest path algorithm to be used to find a route from a starting location to a goal that approximately minimizes probability of detection. Fuel, time and distance constraints are incoporated indirectly, but one strong assumption is made: Detections are assumed independent across network edges and among different radars. A test program is written in FORTRAN and run on a desktop PC using a battery of tests to validate the method. Problems designed with predictable paths having zero probability of detection are solved in a few hundred or thousand seconds for the predicted, optimal paths. Results on problems having non-zero probabilities of detection are less conclusive, but indicate that the method has promise. Errors due to computations and assumptions, as well as their bounds, are discussed, as are recommendations for further model development.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

An automated tool is presented for determining stealthy routes for an aircraft to penetrate enemy airspace, avoid detection by enemy radars, reach a mission objective and return. An existing route-evaluation tool called "TAMPS" (TACTICAL AIRCRAFT PLANNING SYSTEM) can only evaluate the probability of detection of an aircraft flying a route devised by a mission planner. Therefore, planning a stealthy mission currently involves guessing at a "good" route, and iteratively improving that route using guesswork together with TAMPS. This guesswork is unacceptable since the effect of a less-than-optimal route can be the difference between success or failure of a mission.

This thesis develops a prototypic method for automatically determining an optimized flight path for an aircraft from its base to its mission objective that approximately minimizes the probability of detection by enemy radar. The basic scenario posited is that of a Naval search-and-rescue helicopter penetrating an area containing hostile radar installations of known locations and characteristics. However, the methodology is appropriate for most types of air operations subject to detection by hostile radar, although there is no attempt to model potential electronic jamming capabilities of the penetrating aircraft.

The airspace being penetrated is modeled as a three-dimensional network whose vertices are regularly spaced points in the airspace and whose edges represent possible "flight segments" between adjacent points. Using standard radar models and taking

terrain-masking into account, a "length" is assigned to each edge of the graph that is a function of the probability of detection by the radars in the operating area. A shortest-path algorithm is then used to find a route from a starting location to a goal that approximately minimizes the overall probability of detection. Fuel, time and distance constraints are incorporated indirectly, but one strong assumption is made: Detections are assumed independent across network edges and among different radars.

A test program is written in FORTRAN and run on a desktop PC using a battery of tests to validate the method. Problems designed with predictable paths having zero probability of detection are solved in a few hundred or thousand seconds for the predicted, optimal paths. Results on problems having non-zero probabilities of detection are less conclusive, but indicate that the method has promise. Errors due to computations and assumptions, as well as their bounds, are discussed, as are recommendations for further model development.

I. INTRODUCTION

This thesis proposes a prototypic method for quickly selecting a good flight path for an aircraft flying from its base to its mission objective when the flight path must be over a geographic area containing hostile radar installations. This method superimposes a "lattice network" above the region containing the base and objective and defines the edge lengths as a function of detection probabilities. The aircraft is routed through that network so as to reach its objective while approximately minimizing probability of detection.

Aircraft, environmental, and radar models are developed to define the network of vertices and edges and their associated properties. The aircraft considered for this thesis has characteristics resembling those of a naval helicopter. A helicopter is considered, both because of an immediate need for specialized helicopter route planning and because some of the characteristics unique to helicopters simplify the model, e.g., constant fuel burn rate. These characteristics will be discussed later. The environmental model allows the airspace through which the aircraft must fly to be represented as a discrete set of vertices connected by edges. The radar model permits the computation of edge lengths as a function of probability of detection. A shortest-path algorithm is then applied to the network model using the computed edge lengths to determine a flight path that approximately minimizes the probability that the aircraft will be detected by an enemy, without consideration of fuel or other constraints. Adding other constraints to restrict

fuel consumption or distance may make the problem NP-complete, and solving such problems is beyond the scope of this thesis. However, methods to deal with such constraints are discussed in Chapter V. If a route with zero probability of detection is available, the algorithm will select that route. If more than one route is available, a route will be selected using physical distance covered as a secondary measure to be minimized, and route altitude as a tertiary measure to be minimized.

A. MOTIVATION

The motivation behind this thesis is simple: We wish to reduce the chance that our aircrews will be shot down in combat. We propose to accomplish this goal in two ways, by reducing the exposure to the enemy through good route selection, and by reducing the amount of time required for mission planning. By quickly selecting a good flight path, the amount of time and work required to plan the mission is reduced. For emergent missions, such as Search-and-Rescue missions, this will lessen the probability of shoddy mission planning due to time constraints. Should the planner have the luxury of time, less time spent on route planning will allow more time to be spent on other aspects of the mission plan. Currently, the Navy has no automated route-selection capabilities.

The aim of this thesis is to provide a prototype for a route-planning aid that minimizes an approximation of the probability of detection over the mission. With such a tool, the mission planner may quickly provide combat pilots with a stealthy route.

B. BACKGROUND

There are two mission-planning decision aids generally available to Navy mission planners. One converts raw intelligence data into information that is more readily used in mission planning, and the other provides routes for aviation missions. Both of these tools are examined here to determine the feasibility of using them as a means to solve our problem.

Currently the Navy uses one *combat* air route-mission planning aid to assist the mission planner. This system is a high resolution model called TACTICAL AIRCRAFT MISSION PLANNING SYSTEM (TAMPS) (Operational Test and Evaluation Force, 1991). TAMPS takes as input an intelligence database of known enemy positions and equipment (sea- or land-based), and either a) shows enemy radar and weapons coverage zones on a graphic display, or b) generates cumulative probability of detection (P_d) and probability of kill (P_k) figures at designated waypoints along the route. The mission planner can take advantage of the graphic display as an aid to route planning, and then use the P_d and P_k figures to evaluate the result.

TAMPS does not give the planner a way to initiate the route-planning process. The planner must find a path through the radar defense zone, and cannot know whether the proposed route even approaches optimality with respect to overall route P_d and P_k. As TAMPS is an extremely high resolution model, and has a very large and complex database associated with it, it would be difficult to modify it so that it would not only provide worthwhile graphics and high resolution analysis of flight path observability, but also quickly determine a reliable, low-risk flight path.

We also note that Bailey (1992) develops a model that could be used to measure the cumulative time an aircraft is subject to the lethal fire of an air defense network. Time subject to lethal fire is another measure of effectiveness that could be used for route planning but, as with TAMPS, the Bailey model is a "route-evaluation" tool, not a "route-determining" tool. Likewise, a simulation model of Bailey et al. (1994) would only be appropriate for route evaluation.

One *noncombat* aircraft routing aid available to the military is the OPTIMAL PATH AIRCRAFT ROUTING SYSTEM (OPARS) (Systems Research Group, 1966). OPARS uses flight characteristics of individual aircraft such as C-5s or C-130s, and takes navigational fixes, winds aloft, temperatures, etc., and determines the most efficient flight path for the aircraft with respect to fuel consumption, time, and distance flown. As OPARS is principally used for high altitude, low risk flight planning, and does not consider enemy radar or weapons systems, it is inappropriate to use OPARS to plan stealthy routes for combat missions.

Other models and prototypes not in current use have dealt with route planning over a complicated terrain model, and some are briefly reviewed here. Many are not reviewed below because they involve models that are classified, beyond the scope of this work, or are not available for general military use.

1. Application of Previous Routing Models

The models reviewed in this research are different implementations of the same concept: Conduct a search over a search space, i.e., the operating area, and determine the least cost route from the designated starting point to the designated goal,

using a specified metric such as probability of detection. In addition to considering models that have direct applicability to this problem, models that have potential to be extended to encompass this problem are reviewed. Each of the models is highly dependent on the representation of the search space, in particular, the representation of the terrain. As such, we will examine the models from the perspective of how they represent the search space.

a. Terrain and Airspace Models

Several types of terrain representations have been considered as a basis for building the overall model of the environment in this thesis:

- Point elevations,
- Fitted and scaled probability mass functions,
- Fitted polynomial functions, and
- Polyhedral surfaces.

The first representation involves a discretization of the region while the last three are parametric representations. Each representation relies on some source of stored data, where ground elevations correspond to positions on the ground. One obvious source of data is the existing library of prepared paper maps. Another source consists of data files containing ordered ground elevations from a particular area. These files are distributed by the Defense Mapping Agency and are known as Digitized Terrain Elevation Data files (DTED) files. These files will be discussed in more detail later.

The Systems Research Group (1966) utilizes an evenly spaced two-dimensional discretization of the region to represent terrain in the DYNTACS tank

weapons system simulation. Each point in the discretization is represented as a point elevation. This representation is simple to implement using the point elevations and linear interpolation to determine ground elevations between the discretized points. It is also easily extended to include the airspace above by discretizing the airspace in the third dimension as well. One major drawback of this representation is the enormous amount of storage and memory required to handle the data for large geographic regions.

Compared to point elevations, parametric representations of terrain reduce the amount of data needed to represent large geographic areas. Instead of storing each data point in the region, relatively few equations and associated parameters representing the behavior of the terrain are stored. Given a location on the ground, the parametric model computes the terrain elevation at that point by evaluating these equations. There are many ways to parameterize a set of points; we examine three of them here.

Hartman (1979) examines the parametric representation of terrain and its implications on line-of-sight modeling in the STAR combat model. In order to represent hills, Hartman proposes using variations on the scaled bivariate normal probability mass function displaced appropriately on the Cartesian plane. Unfortunately, this is labor-intensive as the fitting of the probability mass functions to terrain features is done by hand. Automation of the terrain fitting procedure might be possible, but is beyond the scope of this thesis. In addition, Hartman indicates technical problems inherent in the representation of rugged terrain as the sum of improper integrals that may not be solved

in closed form. Lastly, it is unclear how this type of terrain model would be extended to provide a model for the airspace above.

Another way to parametrically represent terrain is via a fitted polynomial. In this way, a polynomial $z=F(x,y)$ (where x and y represent map coordinates and z the terrain elevation) is fitted to the ground elevation data points in three-dimensional Cartesian space. It is important to note, however, that in order to capture the "noise" inherent in actual terrain, a polynomial of high degree must be fitted to terrain data of high resolution, such as that provided in DTED files. If, for example, the area to be mapped is a one mile by one mile square (small for our purposes) and this area is divided into 100 yard by 100 yard squares (very low resolution for terrain modeling), then the resulting fitted polynomial would be on the order of degree 16 in both x and y. The time needed to compute these polynomials in a realistic instance, coupled with the degree of error that may result, make these polynomial representations impractical. As in Hartman's model, this terrain representation is not easily extended to include the airspace above.

Zyda, et al. (1986), Diehl, et al. (1986) and Amman, et al. (1986) describe a polygonal representation of terrain. In this model, terrain is represented as a set of partial planes that intersect to form polygonal figures. This is another method of parameterizing terrain, and the equations stored are the equations for the intersecting planes. Though the method may require a large number of equations and much preprocessing of the data for those equations in order to effectively represent a geographic area, this terrain model appears promising for modeling two- and three-

dimensional systems. Also, this model is extendable to include the airspace above the ground. This can be done by extending the planes used to represent the terrain features, so that they partition the airspace above into convex regions of uniform visibility. This extended model has been used by Wrenn (1989) to model the flight path of a terrain-hugging cruise missile. Wrenn further divides each convex region into smaller zones having "equal cost of passage." These costs of passage are computed as a function of P_d, fuel burn rate, distance, and other factors. This model is inappropriate for our use since the airspace regions that would represent regions of reasonably equal probability of detection must be so small, and the corresponding number of equations to describe them so large, that the advantages of a parametric representation over point elevations would be lost. The inappropriateness of Wrenn's method for route planning is discussed in further detail below.

By a process of elimination, we have concluded that the representation of terrain as point elevations is most appropriate for our use. We will use this representation of the region despite its large and inefficient data storage requirements, since any practical parameterization of the region either cannot be extended to include the airspace or will also require large amounts of storage.

b. Radar Models

A model that will approximate the probability that a radar will detect a target is desired so that we may assign lengths to the edges in the proposed network model. Two methods for estimating the probability of detection have been examined: Cookie-cutter detector models and approximations of probability of detection as a

function of time and distance. Enhancements to the cookie-cutter detection model are also examined. Each of these radar models is implementable as our radar model; the task is to determine which model better suits our needs.

A cookie-cutter model in two dimensions sets a fixed detection range from the detector to the target within which detection is certain (Washburn, 1989). We can describe this type of detector as a disk. If a target contacts the disk, then it is detected. The cookie-cutter detector model is simple to implement and it is easy to extend the model to three-dimensions as either a cylinder of certain detection, or as a sphere of certain detection. Further, in lieu of detection certainty, a probability of detection may be associated with the region contained in the cookie-cutter. Of these extended cookie-cutter models, the sphere of constant probability is most appropriate to this thesis. Nevertheless, it does not capture all that we wish to model. The cookie cutter model is not easily modified to permit the use of terrain as a mask and it does not distinguish between a route that passes momentarily through the detection sphere and a route that passes straight through the center. As such, it does not capture the essence of radar avoidance tactics. In a model of any useful resolution, this is unacceptable.

Another enhancement of the cookie-cutter detector model involves modeling the detection range as a random variable (Washburn, 1989), taking into account the random nature of environmental and operator performance degradation. This random variable can then be used in a Monte Carlo simulation. There are two factors which make this model undesirable. The first is the dependence on environmental conditions and operator performance. Though radar performance is affected by environmental

conditions and operator performance, a model for detection that is robust with respect to these effects is desired. The second factor is that a simulation will only return a detection decision. In order to evaluate the probability of detection, many simulations would have to be performed, and it is not clear that we would see any significant effect on accuracy, and would likely see an adverse effect on efficiency.

There are many physical models for radar detection. These models calculate the probability of detection as a function of target, radar, and environmental conditions. We would like to develop a simple, physical model of detection that uses target and radar parameters as constants, is robust with respect to varying environmental conditions, and uses range from the radar and time within the radar maximum detection range as independent variables. Such a model will be developed in this thesis.

c. *Route Planning Algorithms*

An algorithm that will determine the shortest path from a designated starting point to a designated goal will be necessary to find a path that approximately minimizes probability of detection. There are many methods of conducting a search for the optimal route through a network. In this section we review several algorithms that have been implemented to solve problems similar to ours.

Dijkstra's algorithm is a well-known algorithm useful for solving single-source shortest-path problems for graphs with non-negative edge lengths, e.g., (Ahuja, et al., 1993, pp. 108-122). Dijkstra's algorithm can also be modified to stop as soon as a designated vertex in the network is encountered. Modifying Dijkstra's algorithm in this

fashion allows its use as a single-source, single-sink shortest path algorithm. Its ease of use, efficiency, and simplicity make Dijkstra's algorithm ideal for our use.

In his dissertation, Mitchell (1986) proposes a continuous extension of Dijkstra's algorithm for route planning, which he calls the Continuous Dijkstra Paradigm. Mitchell shows the applicability of his paradigm to two- and three-dimensional path planning problems through polygonally modeled search spaces. One such example of an application is the Warehouseman's Problem . Mitchell (1986) applies his algorithm only to regions containing solid impassable obstacles. Mitchell's proposed algorithm does not address the case where the obstacles are passable with increased costs of passage. In addition, the complexity of the algorithm for three-dimensional problems in which the obstacles must be avoided has not yet been determined. Though his proposal is interesting, Mitchell's algorithm lacks definition for our problem and will not be used.

One proposal to optimize an aircraft's route through a geographic region (Wrenn, 1989) utilizes an extension of the polygonal terrain representation described in Zyda, et al. (1986), Diehl, et al. (1986) and Amman, et al. (1986). Wrenn attempts to correct the airspace structure deficiency left in those papers by extending the intersecting partial planes to form regions above the ground. These regions are further divided into convex subregions associated with a constant probability of detection, and are then represented as points. The point chosen to represent a given convex region is the geometric center of the region. Using these points as vertices in a network, a shortest path is computed using an A^* search strategy. The A^* search strategy used by Wrenn is similar to Dijkstra's algorithm except that it uses the sum of two functions to evaluate the

optimality of a route. As in Dijkstra's algorithm, the A^* search evaluates the length of the route from the start point to each vertex. Unlike Dijkstra's algorithm, it then adds to this length an estimated length of the remainder of the route to the goal. This requires some a priori knowledge of the route from each vertex to the goal. Our problem does not include such knowledge. After computing the shortest route within Wrenn's point model, the regional representation is reinstated and the path is perturbed randomly from point to point and region to region using what Wrenn calls "random ray optimization." A form of Snell's laws of refraction is used to refine turns and angles in the path. In this way Wrenn's algorithm generates better routes about the initial A^* optimal route.

Wrenn's method suffers in three areas: 1) generated paths are unreliable when passing from a region of high probability of detection to a region of low probability of detection; 2) he assumes that the probability of detection is constant over an entire region, which is unrealistic for large regions since probability of detection is highly dependent on range to the radar; and 3) due to problems encountered in using Snell's laws of refraction, he restricts the maximum probability of detection in any region to 0.05 in order to avoid erroneous routes (Wrenn, 1989). A probability of detection of 0.05 constrains the size of the regions used in the model, and this constraint negates the advantages of using a polyhedral parameterization of the region. Wrenn's search objectives also emphasize aircraft performance constraints at the expense of low-observability. Though the objective of the route search might be alterable to emphasize stealth, Wrenn's other assumptions described above make his method impractical for our use.

Ong (1990) proposes a model for planning the route of an Autonomous Underwater Vehicle (AUV). The environment in this model consists of distinct cells roughly of the same dimensions as the AUV. The AUV is required to travel from one point to another, transiting from cell center to cell center. Obstacles are modeled as completely occupying one or more contiguous cells, and the AUV must avoid the obstacles. The analogous problem in this thesis is that of forbidding the aircraft to fly into the ground. However, we are also interested in flying through an area where the costs of transiting the area have continuous characteristics. Ong's model may be extended to encompass problems such as that proposed in this thesis, if the AUV is permitted to pass through the obstacles with some type of cost associated with that passage.

2. This Model

The representation of the region employed in this model is direct representation via linearly connected point elevations. There are several clear advantages to this method. First, the structure is simple to implement, and requires little preprocessing. The terrain data is read directly from an external DTED file; the additional structure above the ground required for this project is built by adding altitude to the ground points and appropriately connecting the new airspace point elevations. Second, there are no complicated computations to make. There is no need to fit a high-degree polynomial surface to the points; there is no need to repetitively calculate the equation for fitted planes. Third, it is a simple matter to adjust the resolution of the model through the input process, whereas in the other representations, changing the

resolution requires a complete recalculation of the model due to the required preprocessing of the data. Fourth, a point representation lends itself to utilizing network algorithms to solve route-planning problems.

A major disadvantage of this representation is the size of the storage and memory required to handle such a model. A typical DTED map, covering a one degree of latitude by one degree of longitude area, takes its point elevations every three arc-seconds. This translates to 1200 units on each side, or 1,440,000 data points. For a three degree by three degree area, or approximately 180 miles by 180 miles, the number of data points increases nine-fold, to 12,960,000.

Probability of detection will be calculated as a function of both range from the radar and time exposed to the radar. This method was chosen over the cookie-cutter and random variable models as it allows for better utilization of current route-planning techniques including maximizing the range at which an aircraft is exposed to a radar and terrain masking to limit exposure.

The shortest-path algorithm used to determine the route through the assembled network is based on Dijkstra's algorithm, a well-known, easily implemented shortest-path algorithm. Though other algorithms are applicable, Dijkstra's algorithm is chosen since it is easily modified to provide a single-source, single-sink shortest route.

C. ORGANIZATION

The remainder of this work is devoted to explaining the algorithm discussed above as well as developing its underlying theory. Chapter II characterizes specific model

development and assumptions for the models, including the search space or environment, the synthesis of the search space into a network, and the radar equations used to compute the edge lengths. Chapter III develops the algorithm and tests and validates it in detail. Included in the discussion are input data, data structures, search objectives, network construction, test procedures, and test results. Chapter IV discusses potential errors due to assumptions and approximations. Chapter V describes potential enhancements and suggestions for further work.

II. DEVELOPMENT OF THE COMPONENT MODELS

When modeling a large and complicated system, it is often desirable to simplify the system somehow. A large system can usually be logically divided into smaller components. Similarly, assumptions can be made that reduce the complexity of a system without sacrificing much accuracy. The large and complex problem of precisely portraying an aircraft flying through the air, subject not only to detection by a radar system, but also environmental factors, can also be simplified. The three major components of this system are the aircraft, the environment in which the aircraft is operating, and the radar that is being used to detect the aircraft. The aircraft is modeled as a point object traveling through the region. We model the environment as a three-dimensional network over the ground such that an aircraft transiting the area flies from vertex to vertex along the connecting edges in the network. The radar model is composed of non-communicating individual component radars. This model is required to provide the lengths along the edges in the environment network. Using basic concepts of radar, the probability that a given radar will detect the aircraft at any point can be determined. In turn, functions of these probabilities of detection are used as lengths along the edges. After the two models are combined into a well-defined network model, a shortest-path algorithm can be used to determine a route from a start vertex to a terminal vertex so as to approximately minimize probability of detection.

A. THE AIRCRAFT

The notional aircraft in this work is a helicopter, modeled as an object traveling through space with constant velocity, fuel burn rate, and radar cross-section. These simplifications are somewhat gross, but justified.

The aircraft is considerably smaller than the area (and our special resolution of the area) in which it operates and hence, we assume the aircraft appears as a point. We wish to provide a route for the aircraft that minimizes the risk of detection while permitting the aircraft to complete its mission. We are not interested in optimizing the route with respect to fuel economy or route length. These problems will be considered in Chapter V.

Helicopters are relatively slow aircraft. Their speed is limited by the speed of the advancing blades in the spinning rotors above the aircraft. For this reason, the top speed of most helicopters is below 200 knots. Helicopters fly most efficiently and comfortably at their cruise speed, which is usually near their top speed. When flying at extremely low altitudes (below 50 feet), in a Nap of the Earth (NOE) or a Terrain Following (TERF) flight regime, the helicopter is usually flown at slower speeds due to pilot limitations: It takes time to react to ground obstacles, and the only way to gain reaction time at low altitude is to slow down. In order to capture the different flight profiles peculiar to helicopter operations, we model the helicopter's airspeed as one of two constant velocities. If the aircraft's altitude is such that it is flying in a NOE or TERF profile, we set the airspeed to 40 knots (slow); if it is flying outside these speed-limiting regimes, we set the airspeed to 150 knots (fast). We make this distinction because the speed at which

the aircraft is traveling will affect the amount of time spent on an edge in the network, which in turn, will affect the probability of detection on that edge.

B. THE ENVIRONMENT

The environment is the three-dimensional area described as the region in which the mission is to be planned, containing both the point of origin and the objective, as well as the positions of known enemy radar installations. The area is bounded on the sides by lines of latitude and longitude, and on the bottom and top by the earth's surface and an arbitrary ceiling. The ceiling may be defined by the mission planner as the highest altitude above ground level (AGL) that the mission may be flown, for example the service ceiling of the relevant aircraft.

This region may be envisioned as a box in the positive octant of standard Cartesian three-space. We impose Euclidean distance as a metric in the operating space and then superimpose a three-dimensional rectilinear lattice of discrete points over the area, such that the horizontal and vertical spacing between the points is uniform although horizontal and vertical distances are different. When this lattice is placed over the terrain in the region, it is "pushed down" onto the ground. The imposed three-dimensional lattice over the terrain can be viewed as multiple layers on the terrain representation, each point in a layer corresponding to a point on the ground, the spacing between the layers equal. Note that every point in a layer will have the same absolute altitude above the ground (AGL). Points in the same layer are called "coplanar." Imposing Euclidean distance as a metric will also enable us to, at least partially, constrain the overall length of the route.

Conveniently, the Defense Mapping Agency has mapped most of the world and has stored the information into digitized maps called DTED files. We can partition our operating area into a collection of these digitized maps, providing complete coverage of the area.

1. DTED Data

Digitized Terrain Elevation Data (DTED) is a digitized representation of bald-faced ground elevations of a region with dimensions one degree of longitude by one degree of latitude, or approximately sixty nautical miles by sixty nautical miles. The data in a DTED file are recorded as a single list of point elevation measurements. Elevation measurements begin at the southwest corner of the mapped region and are taken every three arc-seconds (approximately every 100 meters), proceeding in a column northward, until the northern boundary is reached. The list of elevations is continued in a similar fashion starting at the base of the next column, three arc-seconds to the east. This procedure for recording the ground elevations is repeated until the northeastern corner is reached (McGhee, et al., 1987). The latitudes and longitudes where these elevations are measured offer a convenient method with which to draw a grid with arc-second intervals along the terrain.

Elevation figures are stored as 16 bit numbers (McGhee, et al., 1987). Each packed, binary coded DTED file contains approximately four megabytes of information, expanding to approximately 16 megabytes when unpacked. Since the region covered by one DTED file is small compared with most practical areas of operation, the input for a

larger geographic region will have to be pieced together from several DTED files in order to construct the network.

2. The Network

A network representation of the area of operation can be constructed using the lattice structure previously described and connecting its points in the region so that traveling from point to point may be modeled. The terrain model and lattice offer a convenient set of vertices for the network. Both the manner in which the vertices will be connected by edges and the resolution of the network remain to be developed.

a. Complexity

The complexity of a network can be defined in terms of the *branching factors* of its vertices. Ong (1990) defines the branching factor of a vertex as the number of path choices an object transiting a network will have when the vertex is encountered. However, since the branching factor of a vertex is roughly the same as the degree of that vertex, i.e., the number of edges incident to it, we will define the complexity of a network to be the average degree of the vertices in the network.

If the complexity in a network is high, a search algorithm will take longer to run than if the complexity is less. Network complexity can be reduced, though accuracy usually suffers as a result. In deciding how to connect the vertices of our network, it is important to weigh complexity against accuracy (Richbourg, 1987). We will explore reducing network and algorithmic complexity in Chapter V.

Consider a network covering only one DTED file map. If every node in the lattice were connected to every other node, each node would have degree 1,440,000 * L-1, where L represents the number of layers in the lattice. This representation has a high degree of complexity and would not be practical. Therefore, some compromise reducing the complexity of the network while retaining a measure of accuracy must be reached.

In the three-dimensional lattice described above, each point has, at most, 26 nearest neighbors: the 8 surrounding coplanar vertices, and the vertices immediately above and below the vertex and its coplanar neighbors. On a corner or side of the lattice, a vertex will have fewer neighbors. If we connect each vertex only with its nearest neighbors, the degree of any vertex will fall between 7 and 26. In two-dimensional problems, a vertex will have between 3 and 8 nearest neighbors. Reducing the maximum degree of a two-dimensional network to 8 is generally accepted as the point of diminishing returns between time and accuracy (Richbourg, 1987). As our problem is a three-dimensional problem, a higher degree of complexity may be required to achieve a reasonable degree of accuracy. The obvious three-dimensional analogy to the two-dimensional 8 adjacency rule is to use the 26 nearest neighbors mentioned above. As it is unlikely that an aircraft would travel strictly vertically, it is not necessary to include the neighbors immediately above and below the vertex corresponding to the aircraft's position. Hence the maximum vertex degree in the network can be reduced to 24. There are other methods we can use to intelligently reduce the maximum vertex degree in the network; these will be explored later.

b. Spacial Resolution

The data points available in DTED files are spaced approximately every 100 meters laterally. We will use these points as the base vertices for the network, and will add layers above them to serve as airspace vertices. Is this high resolution needed or can it be reduced with a minimal effect on the accuracy of the model?

Using a lower resolution model, essentially deleting elevation data, might result in a flight path that passes through terrain features. This is undesirable. This is more likely in rugged terrain, and less likely in flat terrain. This model will retain the 3 arc-second lateral resolution. A conceptual model that allows variation of the lateral resolution will be discussed in Chapter V.

Spacing between the layers must also be determined, and will be based on the required vertical resolution. The altimeter used in modern helicopters is accurate to within ±25 feet. In view of this, we designate the minimum distance between the layers to be 50 feet. Note that this distance is approximately 1/6 of the lateral spacing. While it may seem desirable to maintain a certain symmetry in resolution of the lattice, helicopter pilots prefer to fly at lower altitudes when possible. Maintaining vertical spacing on the order of 100 meters between the layers would not permit the distinction between very low altitude flight regimes, such as NOE and TERF, and the helicopter's normally higher cruise altitudes. Lack of this distinction could result in a highly inaccurate flight profile. For these reasons, we will use a distance of 50 feet between layers.

c. *Edge Lengths*

If we wish to minimize the probability of detection over the entire route that the helicopter will fly using the network paradigm, we must assign appropriate lengths to the edges in the network. An appropriate edge length might be the probability that the aircraft traveling from one vertex to another along the connecting edge will be detected by a radar in the radar system. Assuming independence of detections along the route (a strong assumption discussed at the end of this chapter) and between radars, the total probability of detection over the route will be a multiplicative model. However, we would prefer to use an additive model with Dijkstra's algorithm. We can easily develop an additive model for minimizing the probability of detection over the route.

Let (u,v) represent the edge from vertex u to vertex v, M the set of feasible routes from source x to destination y, and Q the set of radars. We seek $m \in M$ such that

$$\begin{aligned} P_d(m) &= \min_{s \in M} \{P_d(s)\} \\ &= \min_{s \in M} \left\{ 1 - \prod_{(u,v) \in s} (1 - P_d(u,v)) \right\} \\ &= \max_{s \in M} \left\{ \prod_{(u,v) \in s} (1 - P_d(u,v)) \right\}. \end{aligned}$$

It suffices to find:

$$\begin{aligned} &\max_{s \in M} \left\{ \ln\left(\prod_{(u,v) \in s} (1 - P_d(u,v)) \right) \right\} \\ &= \min_{s \in M} \left\{ -\ln\left(\prod_{(u,v) \in s} (1 - P_d(u,v)) \right) \right\}. \end{aligned}$$

Now,

$$-\ln\left(\prod_{(u,v)\in s}\left(1-P_d(u,v)\right)\right)=\sum_{(u,v)\in s}\left(-\ln\left(1-P_d(u,v)\right)\right)$$
$$=\sum_{(u,v)\in s}\left(-\ln\left(1-\left(1-\prod_{r\in Q}\left(1-P_d^r(u,v)\right)\right)\right)\right)$$
$$=\sum_{(u,v)\in s}\left(-\ln\left(\prod_{r\in Q}\left(1-P_d^r(u,v)\right)\right)\right)$$
$$=\sum_{(u,v)\in s}\sum_{r\in Q}\left(-\ln\left(1-P_d(u,v)\right)\right).$$

We define the cost of the route:

$$W(s):=\sum_{(u,v)\in s}\sum_{r\in Q}\left(-\ln\left(1-P_d(u,v)\right)\right).$$

So we are looking for route $m\in M$ such that:

$$W(m)=\min_{s\in M}\{W(s)\}.$$

The following section develops the calculation of the probability of detection, $P_d^r(u,v)$.

C. RADAR, RADAR EQUATIONS, AND PROBABILITY OF DETECTION

If the lengths on the edges in the network are a function of the probability of detection for an aircraft on the edge, we must develop a method that computes these

probabilities. It is necessary to address some concepts fundamental to radar theory, define the terms used in radar equations, and explain assumptions made in the process.

For the purposes of this planning algorithm, we take the pilot's point of view, which must be pessimistic. We assume that the radars in the system are surveillance or searchlight radars that scan 360° in azimuth and horizon to zenith in elevation. We assume that the total resistance of each radar system is referenced to one ohm. We assume near-ideal environmental conditions for the radars. Further, we assume environmental noise is constant, and that there are no losses in signal strength due to environmental effects, field degradation effects, and system/operator error. We assume that the detector uses pulse integration technology to improve its target detection probability. Further, we assume that the target is stationary while it is being scanned by the radar.

Integration takes advantage of the fact that on each scan of the target, multiple signals may be returned. The detector then stores these returns, and uses a stronger, composite return to make its detection decision. Though there are many methods of accomplishing this, the preferred method is known as *postdetection* (Skolnik, 1980). We assume perfect postdetection integration.

If we assume that from the time that the target enters the radar detection zone until it leaves that the target is stationary, then we need not account for the differences in relative angular velocities. Conversely, if this detail is included in our model, then it becomes more complicated; we must account for the position of every radar beam during its sweep, in which direction and at what rate it is rotating, and when it was turned on.

The probability of detection on an edge is dependent, however, on the amount of time that the target spends on the edge. For this reason, we will assume that the target will be stationary on the edge during its normal edge transit time, and then will "jump" to the next edge. Errors associated with these assumptions will be addressed in Chapter IV. For a more detailed rendering of the radar equations and definitions described below, see Skolnik (1980).

The following terms are defined for use in the development of the radar equations and probability calculations:

P_t: transmitted signal strength (Watts)

A_e: effective aperture of the antenna

G: receiver gain

R_{max}: maximum range of the radar system (nautical miles)

R: distance between the target and the radar (nautical miles)

w: radar beam width (degrees)

s: radar scan rate (rpm)

f: pulse repetition frequency (Hz)

σ: radar cross section of the target (m^2)

A: radar signal amplitude (Volts)

Ψ_0: root mean square (rms) ambient noise voltage

V_T: voltage threshold for detection

S: returned signal (Watts)

S_{min}: minimum detectable signal (Watts)

t: time that the target is within R_{max} of the radar in seconds.

t_s: time it takes the radar to make one full sweep in seconds.

t_w: (dwell) time it takes the radar to cover one beam width w in seconds.

$P_d(R)$: probability the target is detected at range R

$P_i(t)$: probability the target is illuminated by time t

$P_{d|i}(t,R)$: probability target is detected, given target is illuminated

A radar determines the range of an object by the strength of the signal returned from the object as it is received by the radar. The relationship between returned signal strength and range can be defined by several different equations, each differing from the others by one or more related parameters. Which equation is used often depends on which parameters are available for use in the equations. One basic form of the radar range equation (Skolnik, 1980) that uses only target and radar parameters is

$$R = \left[\frac{P_t G A_e \sigma}{(4\pi)^2 S} \right]^{\frac{1}{4}}$$

Rewriting the equation above, and using Ohm's law and the assumption that the system resistance is referenced to one ohm, the returned signal strength in volts can be determined in terms of range

$$A = \sqrt{\frac{P_t G A_e \sigma}{(4\pi)^2 R^4}}$$

Now, given the range of the aircraft from a radar, the returned signal strength can be estimated. The importance of this will become apparent as the method for computing probabilities of detection is developed further.

First, consider the probability that an aircraft is detected by a single radar. There are two processes involved in the detection of a target, illumination of the target and detection of the target once it has been illuminated. Conditioning the probability of detection on probability of illumination, we see that:

$$P_d(R)=[P_{d|i}(t,R)][P_i(t)].$$

1. Probability of Detection Given Illumination

Suppose for now that the target has been illuminated. The probability that the receiver will register this as a detection must be determined. Since we have assumed that pulse integration is perfect, and that each pulse is independent, then the probability of detection in n pulses is 1 minus the probability of no detections in n pulses, or:

$$P(\text{detection, } n \text{ pulses})=1-[1-P(\text{detection, single pulse})]^n,$$

where the number of pulses returned from a single scan of the target is

$$n=\frac{wf}{6s}$$

Now we consider the probability of detection, P_d, for a single returned pulse. Skolnik (1980) shows that

$$P_d=\int_{V_T}^{\infty}\frac{A_f}{\psi_o}\exp\left[-\frac{A_f^2+A^2}{2\psi_o}\right]I_o\left(\frac{A_fA}{\psi_o}\right)dA_f$$

where $I_o(Z)$ is the modified Bessel function of order 0, A is the returned signal strength, and A_f is the amplitude of the filter output.

The above quantity is not integrable in closed form, so we must resort to numerical techniques. Skolnik (1980) shows that a series approximation for P_d:

$$P_d = \frac{1}{2}\left(1 - erf\left[\frac{V_T - A}{\sqrt{2\psi_o}}\right]\right) + \frac{\exp\left[-\frac{(V_T - A)^2}{2\psi_o}\right]}{2\sqrt{2\pi}\left(\frac{A}{\sqrt{\psi_o}}\right)} * \left[1 - \frac{V_T - A}{4A} + \frac{1 + \frac{(V_T - A)^2}{\psi_o}}{\frac{8A^2}{\psi_o}}\right]$$

is valid when $A_f A \Psi_o \gg 1$ and $A \gg |A_f A|$. As such, this model will only be valid when this is the case, and only radar parameters that fit these constraints are used. We note that the parameters for most modern radars fit these conditions.

Since the detection threshold voltage and rms (root mean square) ambient noise voltage are predetermined, we now have a probability function in A, the voltage of the returned signal. Now, since we can describe the returned signal strength A in terms of range of the target from the receiver, and we have an equation for integrating the number of pulses returned in one scan of the target, simple substitution will yield the probability of detection for a single illumination in terms of range:

$$P_{d_i}(R) = 1 - \left(1 - \frac{1}{2}\left(1 - erf\left[\frac{V_T - A}{\sqrt{2\psi_o}}\right]\right) + \frac{\exp\left[-\frac{(V_T - A)^2}{2\psi_o}\right]}{2\sqrt{2\pi}\left(\frac{A}{\sqrt{\psi_o}}\right)} * \left[1 - \frac{V_T - A}{4A} + \frac{1 + \frac{(V_T - A)^2}{\psi_o}}{\frac{8A^2}{\psi_o}}\right]\right)^n .$$

where:

$$n = \frac{wf}{6s}$$

and

$$A = \sqrt{\frac{P_t G A_e \sigma}{(4\pi)^2 R^4}}$$

2. Illumination by a Radar

If the time that the target spends within the maximum range of the radar is less than the amount of time it takes for the radar beam to make one full sweep, then illumination of the target is not certain. Further, if the time spent within the detection zone is greater than the sweep time, though illumination is certain, the probability of the radar detecting the target will increase with every illumination. Assuming independence of each sweep gives *P(detection in c scans)*=1-[1-*P(detection for one scan)*]^c.

Given that the radar scans at rate s, with beam width w, it is easily seen that the amount of time it takes for the radar to traverse the entire 360° arc is

$$t_s = \frac{360 - w}{6s} .$$

a. *Probability of Illumination*

Recall that t denotes the time that the target is available for detection. Assume that $t < t_s$, and hence that illumination is not certain. Here, we will assume that the position of the radar beam when the target enters the radar zone is uniformly distributed. Now we can easily define $P_i(t)$, the probability that the target will be illuminated in t time units thus:

$$P_i(t) = \int_0^t \frac{1}{t_s} dt = \int_0^t \frac{6s}{(360 - w)} dt = \frac{6st}{(360 - w)}.$$

b. *Number of Illuminations*

Suppose instead that $t \geq \frac{360 - w}{6s}$. Now at least one illumination is certain. Then, the minimum number of target illuminations certain in time t, $c(t)$, can be defined by:

$$c(t) = \left\lfloor \frac{t}{\frac{360 - w}{6s}} \right\rfloor = \left\lfloor \frac{6ts}{360 - w} \right\rfloor$$

where $\lfloor x \rfloor$ denotes the smallest integer less than or equal to x. Note that this formulation may allow some "slack" time in the detection zone where illumination is not certain. Let's call this time t', and we see that:

$$t' = t - c(t)\left\lfloor \frac{360 - w}{6s} \right\rfloor.$$

3. Overall Probability of Detection

The probability the target is detected by a single radar can now be written as a function of the target's range from the radar and the amount of time the target spends within the radar envelope:

$$P_d(t,R) = 1 - [1 - P_{d|i}(R)]^{c(t)}\left[1 - P_i\left(t - \frac{c(t)w}{6s}\right)P_{d|i}(R)\right]$$

The problem, however, is to minimize the probability of detection from a system of multiple radar facilities, which is, assuming independence of detections, given by:

$$P_d(u,v) = 1 - \prod_{r \in radars} (1 - P_d^r(t,R))$$

where $P_d^r(t,R)$=probability of detection with respect to radar r.

D. SUMMARY

By integrating each of the component models developed above, it is possible to model a helicopter transiting a region subject to radar coverage. The airspace through

which the helicopter flies has been discretized, and the points of this discretization can be considered vertices. These vertices are then connected selectively by edges. The helicopter moves through the airspace, from vertex to vertex, along connecting edges. The distance between the helicopter and any radar can be measured, as can the time that it takes for the helicopter to move along the path between two points. Based on the range to the radars in the system, and the time spent on the edge, a probability of detection can be assigned to each edge. Thus, an undirected graph with non-negative edge lengths is well-defined. Now that this network is well-defined, a search algorithm can be applied to find a path that approximately minimizes cumulative probability of detection when moving between origin and destination vertices.

We note that while the assumption of independence of detection probabilities is necessary in order to develop a probabilistic model, it will introduce some error causing the calculations to be somewhat optimistic.

For example, consider a flight path that will take a helicopter just inside the detection zone of a single radar, and that the aircraft will be exposed to the radar for 500 meters (approximately 25 seconds). Applying our probability detection model for the entire distance of exposure results in a nearly certain detection (P_d=0.999999...), while breaking up the route into 5 distinct legs results in a slightly less than certain detection (P_d=0.9945).

The probability of detection over each route is subject to the same type of error, so each route will tend to maintain its *relative* goodness as compared to the other routes,

despite this assumption. The impact of this and other errors incurred will be discussed in further detail in Chapter IV.

III. MODEL IMPLEMENTATION AND VALIDATION

In this chapter the component models for the operating environment and radar system are assembled into a network model and a shortest-path algorithm is applied to various test cases to determine the stealthiest route through the region. This process will be developed as a series of steps: 1) definition and use of the data structure for the network, 2) construction of the network as a set of vertices and connecting edges from the DTED input data, 3) assignment of lengths to the edges, and 4) the search strategy of the shortest path algorithm. A series of tests and results is then presented, followed by conclusions.

A. ADJACENCY LISTS

Two different data structures are commonly used to represent a network model, adjacency matrices and adjacency lists (Ahuja, et al., 1993, pp. 34-35). Both are appropriate for the static nature of our network, but the adjacency matrix would be very inefficient: Adjacency matrices are reasonable only for a small or very densely connected network which our network is not. Thus, an adjacency list representation will be used. The adjacency list structure can be implemented in several ways. The one chosen for our problem will be called a Hierarchical Adjacency List (HAL).

Since the network is not expected to change, we could simply store the vertex information, and algorithmically compute the nearest neighbors to be considered for the next vertex in the route. We have chosen a more complete structure to accommodate

potential future improvements and because it allows modularization of the algorithm which simplifies coding and validation..

The HAL is a two-part structure that consists of two arrays. For each node I, $Ptr(I)$ is the location in array $Adj()$ of the first node adjacent to node I, and $\{Adj(Ptr(I)), Adj(Ptr(I)+1), \ldots, Adj(Ptr(I+1)-1\}$ are all the nodes adjacent to node I. The lengths of the edges will not be stored in an array, as they will be computed "as needed." Computation of edge lengths will be discussed later in this chapter.

B. BUILDING THE NETWORK

As previously discussed, the data points in the DTED files serve as a base of vertices in the network. These vertices represent the terrain of the area of operations. The vertices representing the area's airspace are built upon this base. After the vertices of the network are constructed, they must be connected by the network edges and placed in the HAL. Initially, the edges will be constructed without lengths. This omission is purposely made to save the complex calculation of many edge lengths. Lengths will be appropriately added to the edges later. As the edges are inserted into the developing network structure, the vertices are stored in the HAL described above.

1. Construction of the Vertices

As discussed in Chapter II, the number of entries in and the organization of the DTED files is known. Therefore, as the elevation information is read from the DTED file, each of the terrain vertices can be both enumerated and assigned an ordered triple (x,y,z). It is convenient to enumerate the terrain vertices in the same order as their

elevations listed in the DTED file. The (x,y,z) ordered triple is assigned so that (x,y) represents its position in the Cartesian plane and z represents its elevation. The origin of the plane is defined as the southwest corner of the region. In addition, as each terrain vertex is created, all of the airspace vertices above it can also be created by incrementing the z coordinate of the (x,y,z) triple. The airspace vertices can be enumerated layer by layer using the same ordering used in enumerating the terrain vertices. This is how the lattice structure laid over the terrain, discussed in Chapter II, is built.

If the vertices in the network are enumerated layer by layer, following the same ordering within each layer, then the enumeration of a vertex in each successively higher layer will be a direct translation of the vertex number of its corresponding terrain vertex. In addition, since the vertices are enumerated in a specific order, it is possible to relate the vertex number given by enumerating the vertices to the (x,y,z) ordered triple. We also note that with some adjustments in constants to account for unit changes, this relationship holds if (x,y,z) is measured in arc-seconds, meters, or as latitude and longitude positions. This relationship between coordinate position and enumeration becomes important, for example, when we are translating vertex numbers in the shortest route to map coordinate and altitude waypoints for use in flight.

2. Connecting the Vertices

As the vertices are created, they are connected by edges. Since there is a direct relationship between the number of a vertex and its position in the network, and each vertex will have only its nearest neighbors adjacent, it is a simple matter to compute

the vertex numbers of each nearest (geometric) neighbor, and then place this information in the HAL appropriately.

C. COMPUTING EDGE LENGTHS

The length associated with an edge in the network is based on the probability that an aircraft will be detected by a radar while it is traversing the edge. We consider that the aircraft is traversing the edge (u,v), and that the length $W(u,v)$ assigned to (u,v) is a function of the probability of detection $P_d(u,v)$: $W(u,v)=-\ln(1-P_d(u,v))$. There are several factors that must be computed in order to determine $P_d(u,v)$. They include the number of radars involved in the surveillance, time spent on the edge (u,v), range to each of the radars, and whether or not a line-of-sight between the aircraft and each of the radars exists. The method discussed below involves the simplified case where there is only one radar. This method is then extended so that the aggregate effects of several independent radars can be described using the equation

$$P_d(u,v) = 1 - \prod_{r \,\in\, radars} (1 - P_d^r(u,v)),$$

where $P_d^r(u,v)$=probability of detection on (u,v) with respect to radar r.

The following single-edge, single-radar example illustrates the fundamental procedure for assigning a length to an edge. Suppose the aircraft is traversing edge (u,v), and that there is only one radar located at a terrain vertex s.

First, we determine the range from the aircraft to the radar. We calculate the Euclidean distance between u and s, and between v and s. The range of the aircraft along

(u,v) is taken to be the average of the two ranges. We use the range figures for each vertex to apply a correction to the edge transit time used in the calculation of $P_d(u,v)$. We assume that if only one of the two vertices on the edge is within the radar's maximum detection range then the aircraft is in range on only one half of the edge, and therefore the time the aircraft spends within the radar's maximum detection range is one half of the time spent on the edge. If both are out of range, then the amount of time that the aircraft is available for detection is zero.

Second, we determine whether a line of sight exists from the aircraft to the radar. To accomplish this, lines of sight for each of the two vertices u and v are considered. In order to determine the existence of a line of sight between a vertex and the radar, we subtend lines from both u and v to s. We then "walk" along these sight lines and compare the altitude, i.e., z-value, of the projected line to the elevation of the surrounding land features whenever a lattice-line is crossed. Where the sight line does not intersect a vertex exactly, we interpolate the terrain elevation from the nearest land features, both latitudinally and longitudinally. If the height of the sight line drawn between the vertex and the radar is below the estimated elevation of the land features anywhere along the line then a line of sight does not exist. As in the approximation for time made with respect to range from the radar, we make some adjustments to the time spent within sight of the radar where it applies. If both vertices u and v have a clear line of sight to s we assume that the whole edge has a clear line of sight and the aircraft is exposed during the entire time spent on the edge. If only one vertex is obscured, we assume that the aircraft is exposed on one-half of the edge and reduce the time exposed by one-half. If both

vertices are obscured from the radar, then we assume that the aircraft is not exposed to the radar at all on (u,v). This approximation is justified provided the distance between u and v is relatively small compared to the range to the radar. This should typically be true since the distance between vertices is less than 150 meters, and the distance to the radar will be many miles in any reasonable flight path.

Also, if one of the vertices is out of range, and the other is obscured, then we consider the time that the aircraft is available for detection to be zero.

After the time on (u,v) and range from the radar are computed, the probability of detection $P_d^r(t,R)$ is computed using the equations developed in Chapter II. The single-radar length associated with edge (u,v) is $W(u,v)=-\ln(1-P_d^r(t,R))$.

If there are multiple radars, the single radar length for (u,v) is calculated for each radar and, assuming independence of radars,

$$P_d(u,v) = 1 - \prod_{r \in radars} (1 - P_d^r(t,R)),$$

gives the total probability of detection on the edge (u,v) (see Chapter II). The length of (u,v) is then $W(u,v)=-\ln(1-P_d(u,v))$.

As always, when simplifying assumptions or approximations are made some error may result. In the above estimation of the length of a single edge, several such actions were taken. The errors induced by these approximations will be addressed in Chapter IV.

D. DIJKSTRA'S SHORTEST-PATH ALGORITHM

Now that the network has been built and stored in a HAL, we can use Dijkstra's algorithm to solve this single-source, single-sink, shortest-path problem. Dijkstra's algorithm is a well-known 'label-setting' algorithm useful for solving shortest-path problems for graphs with non-negative edge lengths (Ahuja, et al., 1993, pp. 108-122). This section briefly discusses the implementation of Dijkstra's algorithm for this problem.

As the computation of the length associated with each edge is complicated, and given the large number of edges in the network, the lengths associated with the edges are computed as needed. This is done to reduce the number of edge length calculations that must be performed. At most, the length on every edge will need to be computed once. However, since Dijkstra's algorithm can be modified slightly to halt before every edge in the network is encountered, 'on demand' computation of edge lengths may reduce the number of calculations required. For a more detailed discussion of modifications of Dijkstra's algorithm see (Ahuja, et al., 1993, pp. 108-112).

In the event that two or more routes have the same minimum probability of detection, our algorithm breaks ties using secondary and tertiary keys. The secondary key is distance to the mission objective vertex. The algorithm will include the vertices that have the shortest straight-line distance to the goal vertex. This should have an effect similar to picking the shortest distance route. The tertiary key is vertex altitude. The algorithm favors vertices that have the lowest altitude, in keeping with NOE or TERF flight profiles.

Given the locations where the aircraft will begin and end its mission, our algorithm translates those locations to the nearest terrain vertices in the network. The vertex corresponding to the starting location is called the source vertex. The ending vertex is called the sink vertex. Starting at the source vertex, Dijkstra's algorithm traces through the network and labels each vertex encountered with its associated length computed with the method developed in Chapter II. As soon as the sink vertex is permanently labeled (and, we hope, before a host of other irrelevant vertices are labeled), the algorithm terminates. The length of the route from source to sink is the length associated with the label of the sink, and that length is minimal.

E. ALGORITHM TEST AND VALIDATION

In order to completely validate and verify any model, some test procedure must be developed that can correlate predicted and observed behavior. Obtaining actual observations would involve flying actual aircraft through a real radar network. Both the extent and expense of such a test procedure are beyond the scope of this work. Thus, a complete verification of this model will not be accomplished. However, a computer program implementing the described data structures and algorithm, is written using Microsoft FORTRAN (v5.1) (Microsoft Corp., 1991), and that along with appropriate input data serves to validate the model. This is accomplished by adjusting the input data in such a way as to make the best flight path predictable. If the program consistently produces the predicted flight path then we consider the model to be valid. The program is run on an Intel 486-33, equipped with 4 megabytes of random access memory. All

tests are run from within the Microsoft Windows environment. Table 1 summarizes the test cases and results. All tests and results are presented in more detail in the subsequent sections.

# radars/placement	Terrain	P_d=0 avail.?	P_d found	Time to complete (seconds)
No radar	Rolling	Yes	0.0	2.6
No radar	Rugged	Yes	0.0	131.8
Single radar, centrally placed	Rolling	Yes	0.0	684.4
Single radar, centrally placed	Rugged	Yes	0.0	793.8
Two radars, centrally placed, gap between	Rolling	Yes	0.0	205.2
Two radars, centrally placed, gap between	Rugged	Yes	0.0	244.5
Multiple radars (8) in a diagonal band that completely blocked path, no gaps	Rolling	Yes, must climb to get there	0.0	6428.4
Multiple radars (7) in a diagonal band that completely blocks path. Radar and environmental parameters altered to provide low-cost route.	Flat	No	None	Does not apply, test would not complete; adjusted radar parameters violate model assumptions

Table 1: Summary of Algorithm Test Cases and Results.

# radars/placement	Terrain	P_d=0 avail.?	P_d found	Time to complete (seconds)
Multiple radars (7) in a diagonal band completely blocking path. Probabilities of detection assigned.	Flat	No	0.8	1387.5
Multiple radars (8) in a diagonal band that completely blocked path, no gaps	Rugged	Yes, must climb to get there	0.0	893.2
Random number more than 4 but less than 15, placed randomly. 20 iterations run.	Rolling	Unknown when test begins	0.0	Average: 4783.0
Random number more than 4 but less than 15, placed randomly. 20 iterations run.	Rugged	Unknown when test begins	0.0	Average: 10423.4
Complete coverage of region	Rolling	No	1.0	More than 24 hours
Complete coverage of region	Rugged	No	1.0	More than 24 hours

Table 1: Summary of Algorithm Test Cases and Results. (Cont'd.)

1. Test Data

Several sets of data are needed as input to the algorithm. Terrain data is needed to construct the geographic network, aircraft data is needed to determine aircraft

characteristics and capabilities, and radar data was needed in order to determine radar capabilities.

a. Terrain Data

As we observed earlier, a complete DTED map will require 16 megabytes of RAM. Creating a data structure this large for prototyping purposes is unnecessary; if the algorithm will work on small but complete models, it should work on models of larger scale. In order to test this prototype conveniently on our modestly equipped desktop PC, we used smaller files of terrain data that would only take up approximately four megabytes of space. As such, the small size of the (plane) area used for test purposes is 20 vertices by 20 vertices. Three sets of terrain data were developed to explore various aspects of the algorithm: flat (no elevations), hilly, and mountainous.

Flat terrain was initially used to test the program's overall function, and to test the algorithm's ability to avoid radar without concern for line of sight and other terrain related issues. Rolling terrain tests the algorithm's ability to provide climbing or descending routes, and allows for some terrain-masking. Rugged, mountainous terrain allows the algorithm to make full use of terrain masking when available (Figs. 1, 2), tests the ability to make sharp climbs and descents, and tests for equally good routes that differ only in length.

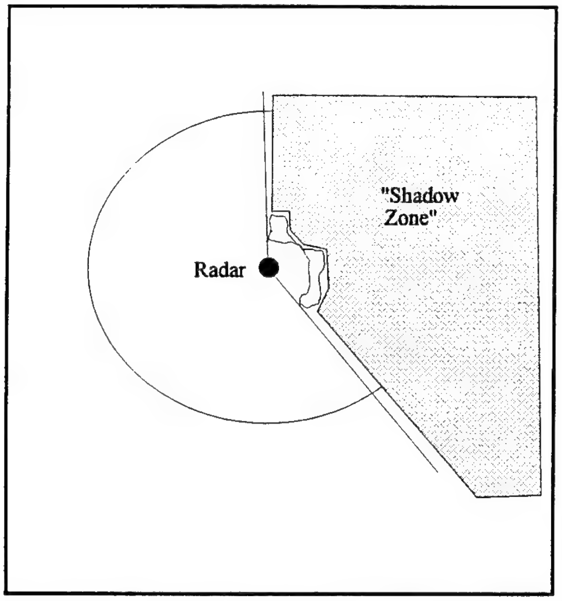

Figure 1: Terrain masking: land features block radar, permitting aircraft to fly through "shadow zone" undetected.
Top view

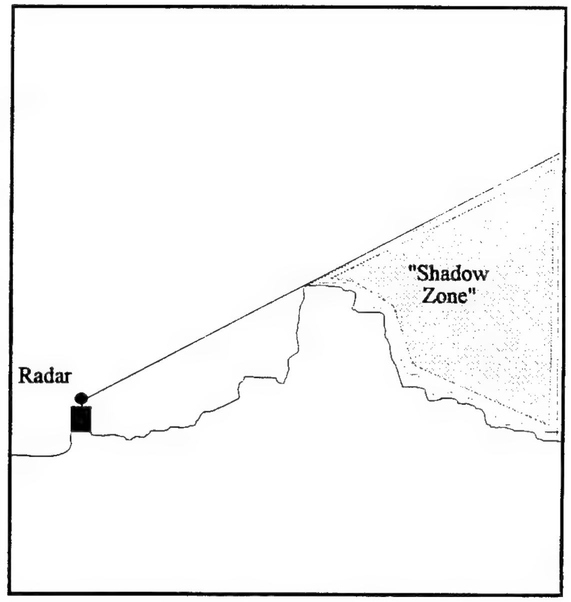

Figure 2: Terrain masking: Land features block radar, permitting aircraft to fly through "shadow zone" undetected.
Side view

b. Aircraft and Radar Data

Aircraft transit speeds and radar reflectivity figures have been made up of composite figures found in unclassified publications (Skolnik, 1980, Taylor, 1989). Actual figures are not used, in order to keep this work unclassified. However, these figures are representative of those for medium-sized military helicopters found world-wide. Likewise radar performance parameters such as power, maximum range, and detection threshold are representative of mobile air search radars systems available on the international market.

2. Test Strategy and Results

A total of fourteen tests have been conducted, each a variation on either the terrain type used or number and location of radar. The basic test procedure is to set the terrain type and within that terrain type change the number and configuration of the radars, starting with no radars, and finishing with every vertex covered by at least one radar. By first using no radars so that the probability of detection is zero, the algorithm's shortest path should be the shortest distance route. The number of radars is then gradually increased to test the algorithm's ability to sense and avoid radar, its ability to choose a lesser probability route by increasing the route's physical length, and finally its willingness to allow the aircraft to fly through a radar coverage zone when no other option is available. The tests and results are grouped into total number and arrangement of radar stations. The tests and associated results for rolling and rugged terrain are presented below. Occasionally we find that it takes less time to select a route over rugged terrain. This occurs whenever land features obscure radar coverages, allowing for a more direct, no-cost route.

a. No Radars

In both terrain cases the algorithm chooses the lowest flight path with the shortest physical length. This is as it should be, as the probability of detection on each edge is zero, the secondary and tertiary priorities (shortest route and low altitude) become more important. The entire route selection process took 2.6 seconds for rolling terrain and 131.86 seconds for rugged terrain.

b. Single Radar

With a single radar placed to cover a direct flight path from start to finish, the algorithm chooses the shortest, lowest path that completely avoids the radar. When the radar is centered on the direct path from start to finish in the case of flat terrain the algorithm invariably chooses the northerly route (Fig. 3). This is due to the way the algorithm is instructed to choose the next vertex in the network. This decision criterion is easily alterable, and has no bearing on the accuracy or quality of the model. Placing the radar slightly off-center will result in the shortest distance route being selected, as was demonstrated in the multi-radar cases. The entire route selection process took 684.43 seconds for rolling terrain and 793.8 seconds for rugged terrain.

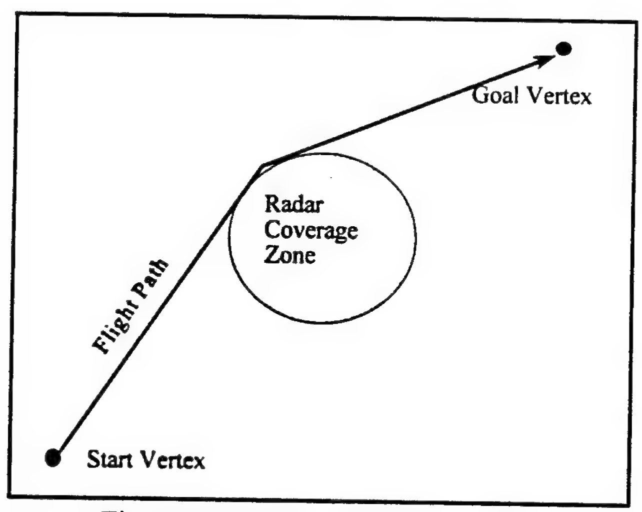

Figure 3: Single radar flight path

c. Two Radars

Two radars were placed so as to provide a narrow gap through which an aircraft could fly directly from the start vertex to the finish vertex. This is exactly what

happened (Fig. 4), with the rare exception when using a mountainous terrain data set (Fig. 5). Occasionally, when the terrain is extremely rugged, the algorithm sacrificed the physical length of the route in order to provide a lower altitude, no-cost route. This is acceptable as the performance of helicopters improves at lower altitudes. This tactic is

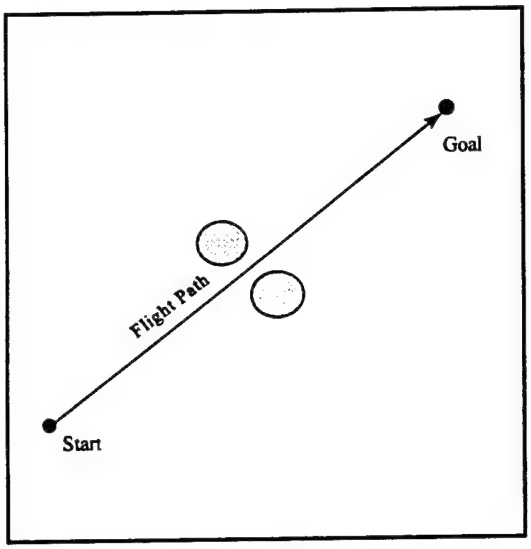

Figure 4: Gap provides direct route to goal

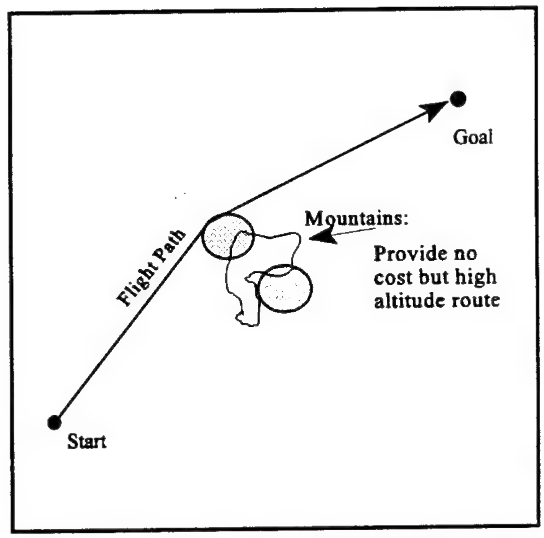

Figure 5: Given two no-cost paths, low-altitude route chosen

in keeping with current radar avoidance doctrine. The entire route selection process took 205.22 seconds for rolling terrain and 244.52 seconds for rugged terrain.

d. Multiple Radars

Several radars were placed in a band across the region in such a way that the radar coverage zones incompletely overlapped (Fig. 6, 7) or did not overlap at all; no radar was located within the radar coverage of an adjacent radar. If a low altitude no-cost route was available due to terrain masking, the program selected that route. Otherwise, the flight path climbs as much as necessary to avoid flying through a radar coverage zone, utilizing the valley created by the intersecting spheres of radar coverage. The entire route selection process for non-overlapping radars took 4164.63 seconds for rolling terrain and 1085.81 seconds for rugged terrain. The entire route selection process for overlapping radar coverage took 6428.42 seconds for rolling terrain and 893.23 seconds for rugged terrain.

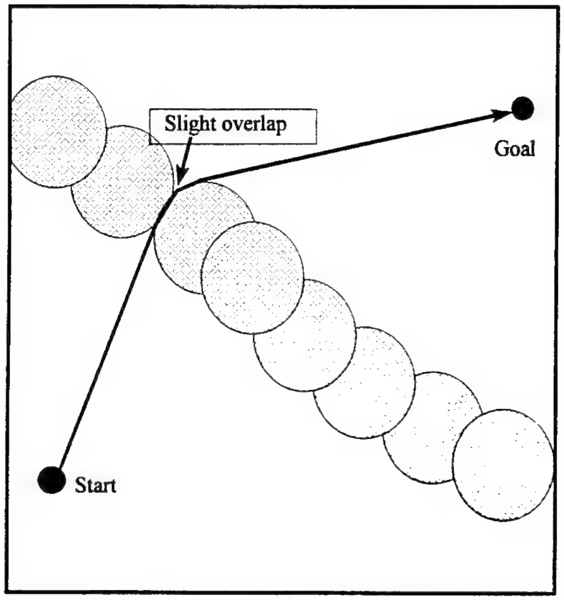

Figure 6: Aircraft picks lowest no-cost route over radar "valley" created by overlapping coverage. Top view

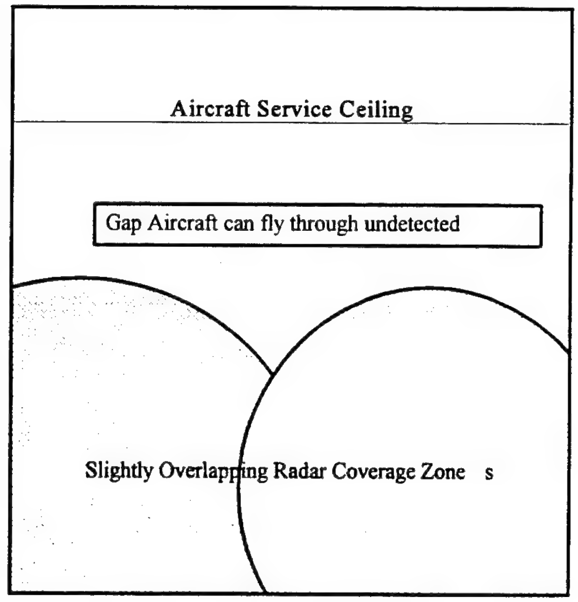

Figure 7: Aircraft can fly through the radar "valley" created by overlap. Side view

Once again, a series of radars was placed diagonally across the region completely blocking access from the start vertex to the finish vertex, with some changes in parameters: the service ceiling was lowered to prevent the aircraft to climb and fly through the gap in coverage as in previous tests, and the radar parameters were altered to provide coverage that would permit exactly one illumination on one series of edges

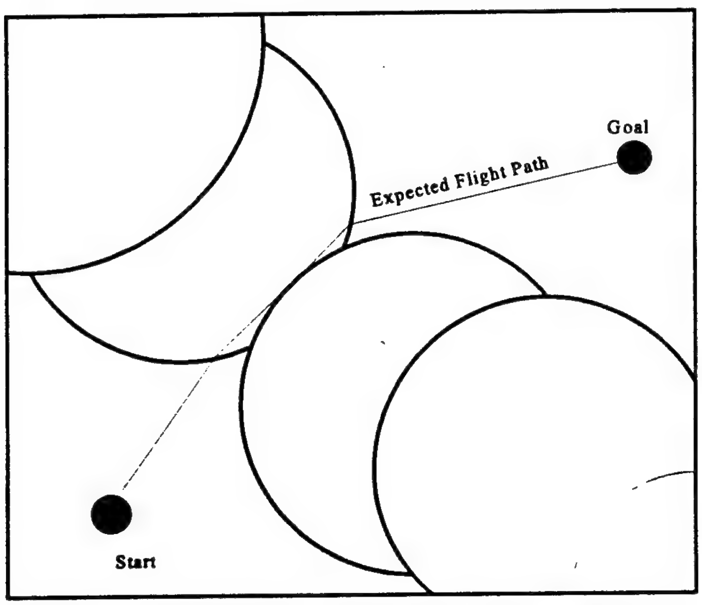

Figure 8: By adjusting ceiling and radar parameters, a low-cost route is forced

traversing the radar coverage zone (Fig. 8). By just skirting a single radar coverage zone, the probability of detection should have been somewhat less than certainty. However, using these altered and unrealistic radar parameters, this test was unable to provide a reasonable probability of detection. We suspect that by adjusting the radar parameters our model departed from realistic radar parameters and one or more of the assumptions

integrated in the model were violated, invalidating the radar model for use in this instance.

In order to test the algorithm without encountering the above difficulties, we assigned probabilities of detection to edges based on their exposure to radar: P_d=0.0 was assigned to edges that were not exposed to radar, P_d=0.1 was assigned to edges that were half-exposed to radar, and P_d=0.3 was assigned to edges that were completely exposed to radar. Only flat terrain was used to eliminate any variations associated with radar masking. Cases of multiple radar exposures was treated as in all other tests. The radars were distributed such that there were heavy overlaps in all but one of the radar coverage zones. The algorithm chooses the expected path, selecting a route that maximized the distance from the one radar and ensuring that there were not any multiple exposures. The route selection process took 1387.56 seconds, and results in a probability of detection of 0.8055190.

In addition, a random number of multiple radars were randomly placed about the region to test the adaptability of the algorithm. While there is no predictable outcome to this test, the algorithm provided a seemingly reasonable flight path, each time finding a no-cost route. The entire route selection process took on average 4783.02 seconds for rolling terrain and 10423.46 seconds for rugged terrain.

e. *Complete Coverage*

To ensure that the algorithm would choose a route of certain detection when no other options were available, the test region was saturated with radar units such that each vertex was covered by more than one unit. This is an extreme and very unrealistic case, and these tests took a substantial and impractical amount of processing

time. However, in each case a low-altitude route with certain detection was chosen. This test took overnight for both rolling and rugged terrain cases. Exact times are not available.

3. Summary

For each of the test cases and each terrain type, except as noted, the program runs as predicted. Some errors were detected, probably within the series approximations used in the radar equations, but with some additional work, these could be corrected or more accurate approximations derived. We propose that the model developed will provide a stealthy route through a radar network, and that further work be conducted to make this a fully functional, useable product.

IV. ANALYSIS OF ERRORS

As in any model of an actual system, there are errors introduced as the model is formulated, and this model is no exception. However, if the error involved can be bounded, then we can expect the predicted or modeled results to approximate reality within a certain degree of accuracy. Moreover, this information can be used to decide where to add or subtract detail in the model based on the contributions these changes make to the accuracy of the model. In general, sources of error originate from: a) assumptions made, b) approximations used, c) error inherent in the algorithm, and d) roundoff error when computations are made. This chapter addresses those errors caused either by our assumptions or our approximations.

A. ERRORS FROM RADAR MODEL ASSUMPTIONS

1. Independence Assumption

In developing the probability of detection equations in Chapter II, it is necessary to assume that detection on coincident edges in a flight path are independent of each other. Given the relatively short physical edge length, this is not necessarily true, especially if contiguous edges in a route are all exposed to the same radar. The effect of the independence assumption is that edge lengths for several short exposure times are computed instead of one edge length for the actual, longer exposure time. The net result is that the probability of detection calculated over the route is unduly optimistic. We see no way to eliminate this assumption without either overly complicating the model or

rendering the computation of edge lengths intractable. Nor can we quantify the error induced by the assumption of independent edges. We can say, however that the route selected is still a good route relative to all other routes, but absolute quality of a route cannot be judged. As such, this model is currently only useful in planning a mission that must be flown, and any probability of detection figures obtained should not be used to decide whether or not the mission should be attempted. However, if the selected route were further processed into larger straight line segments, it could be evaluated using TAMPS for a more accurate probability of detection.

2. Stationary Target Assumption

If and when the assumption of independence of detection among edges can be relaxed, the following analysis will be important for bounding the overall probability of detection.

The assumption that the target is stationary while it is illuminated affects the computation of the target's illumination time, and hence the probability of detection on that edge.

Recall the following terms:

s: radar sweep rate (RPM)

w: radar beam width (degrees)

R: range to radar (nautical miles)

We can determine the radar beam radial velocity at range R, as

$$\Theta_s = (2\pi R s)\left(\frac{1\text{ min}}{60\,seconds}\right) = \frac{\pi R s}{30}$$

since the distance in one revolution is $2\pi R$, and the radar beam makes s revolutions per minute. The distance covered in one illumination is the length of the arc subtended across a wedge of radial width w and distance R, or $\frac{w}{360}(2\pi R)$. Therefore the amount of time that a stationary target is illuminated is:

$$t_{stationary} = \frac{\frac{w}{360}\,(2\,\pi\,R)}{\frac{\pi\,Rs}{30}} = \frac{w}{6s}.$$

Suppose that the target is moving during the time that it is being illuminated. Then, the time it is illuminated actually depends on its velocity relative to the sweeping radar beam. We can break this problem into two cases: the target is moving against the direction of the radar sweep, and the target is moving with the direction of the radar sweep.

a. *Target Moving Against Radar*

Now let v denote aircraft speed in nautical miles per second. In addition, assume that the velocity of the aircraft closely approximates its radial velocity relative to the radar for the short time it is illuminated. Although the aircraft is traveling in a straight line, we will also assume, because of the relatively small beam width w and the rotation of the radar, that the aircraft flies along an nearly circular arc relative to the radar beam during illumination. As the only factor that has changed from our stationary illumination time is the relative velocity of the aircraft with respect to the radar, we can easily compute the actual amount of time that the target is illuminated, and we now say

that the amount of time that a target moving with velocity v, against the direction of the radar sweep, is illuminated is:

$$t_{moving} = \frac{\frac{w}{360}(2\pi R)}{\frac{\pi R s}{30} + v} = \frac{w \pi R}{6 \pi R s + 180 v}.$$

b. Target Moving with Direction of Radar

Similarly we can develop a model for the amount of time that an aircraft traveling in the same radial direction as the radar beam. Since the radar beam is still moving at the same rate, and the distance between the leading and trailing edges of the beam are the same, the models are identical, with the exception that the relative velocity is subtractive rather than additive, so:

$$t_{moving} = \frac{\frac{w}{360}(2\pi R)}{\frac{\pi R s}{30} - v} = \frac{w \pi R}{6 \pi R s - 180 v}.$$

c. Bounding the Error Incurred

By calculating $\frac{t_{stationary}}{t_{moving}}$, we can easily see that the percent relative error created by assuming that the target is stationary on an edge is $t_{\Delta} = 1 \pm \frac{30 v}{\pi R s}$ %, depending on whether the aircraft is flying with or against the radar beam.

This illumination time effects the number of radar pulses that will strike the target, and hence will effect the single scan probability of detection. The relative error in n, number of pulses per illumination is $n_{\Delta} = \pm \frac{30 v f}{\pi R s}$ %.

Now, recall that

$$P_d^r(R) = 1 - \left(1 - \frac{1}{2}\left(1 - erf\left[\frac{V_T - A}{\sqrt{2\psi_o}}\right]\right) + \frac{\exp\left[-\frac{(V_T - A)^2}{2\psi_o}\right]}{2\sqrt{2\pi}\left(\frac{A}{\sqrt{\psi_o}}\right)} * \left[1 - \frac{V_T - A}{4A} + \frac{1 + \frac{(V_T - A)^2}{\psi_o}}{\frac{8A^2}{\psi_o}}\right]\right)^n$$

and that

$$P_d\ (t,R) = 1 - [1 - P_{d|i}(R)]^{c(t)}[1 - P_i\ (t - \frac{c(t)w}{6s})P_{d|i}(R)].$$

This error only affects the single scan probability of detection in the last term of the series approximation and is in turn raised to a positive integer exponent greater than one. As such, we suspect that this error is insignificant. To further investigate its effect, pessimistically assume that $P_{d|i}(R)$=0.5. The relative error of a single-radar, single-edge probability of detection due to the stationary assumption can now be bounded as being not more than $[0.25]^{\frac{30vf}{\pi Rs}}$ % (less than one-quarter of one percent). We consider this error to be insignificant.

3. Other Assumptions

There are several other assumptions made in Chapter II in order to simplify the detection models. These assumptions regard environmental factors, aircraft characteristics, or radar parameters.

a. Environmental Factors

In Chapter II we assume that noise, signal attenuation, and other relative environmental conditions are not affected by random events. By using a function of P_d to judge the relative, as opposed to the absolute, quality of the route, the dependency of

the model on environmental factors can be eliminated. Eliminating the random nature of environmental effects simplifies the probability calculations and permits us to develop an analytic model. By introducing environmental noise as a random phenomenon we would be required to change the model from analytical in nature to a simulation. As such many simulations would be required before a reasonably accurate route selection could be made. A simulation would not be significantly more accurate than this model, and it would take considerably longer to complete the process.

b. Aircraft Characteristics

Several assumptions are made regarding target size and performance, but the most important assumption made concerns its radar cross-section. By assuming a constant radar cross-section, independent of the aspect that the aircraft presents the radar, we have eliminated one of the controllable variables in the problem and simplified the probability calculations. This assumption also eliminates the entire superset of route configurations that are aspect-dependent. Obviously, any pilot should want to present the smallest radar cross-section to each radar. It should be just as obvious that this is very impractical in flight. Assuming a constant cross-section eliminates this unneeded complication in the planning process and leaves in-flight adjustments to the pilot's judgement.

c. Radar Parameters

In calculating the probability of detection on any given edge, we make several assumptions regarding the relevant radars: 360° and horizon to zenith coverage,

no communication between radars, and no operator error. With the exception of non-communicating radar stations, these assumptions are completely in keeping with our desire to be pessimistic and conservative in our calculations. To assume noncommunicating radar stations is consistent with the assumption of independence treated earlier, though it is not consistent with our desire to be pessimistic.

B. ERRORS FROM APPROXIMATIONS

During the development of the model and implementation of the test algorithm, several approximations are made in the actual computation of the probability of detection or the parameters that are involved in that calculation, namely time and distance.

1. Error from Series Approximation for the Probability of Detection

The series approximation used in Chapter II to find the single-scan probability of detection was drawn from Skolnik (1980), the recognized standard text for radar and radar systems design. The aim of this thesis is not to break new ground in the realm of radar applications, but rather to use current information in a new and interesting way. Any error generated by this approximation is acceptable for our purposes.

2. Error from Range and Time Approximations

Recall that in applying the radar model several approximations are made: a) the location of the radar as the terrain vertex nearest its actual location, b) the range from the aircraft to the radar as the average range from the edge being traversed, and c) the time spent exposed to the radar as zero, one half, or all of the time spent on the edge, depending on whether both vertices on the edge are in range of the radar and whether

both vertices had a clear line of sight to the radar. The following sections will address the errors associated with these approximations.

a. Error from Approximating Radar Location

During the input phase of the test algorithm, the position of the radar stations is translated from latitude and longitude to vertex coordinates. In doing so, it may be necessary to alter the position slightly in order to situate a radar station precisely on an existing vertex. The error involved in such an approximation becomes greatest when the actual location of a radar is equidistant from four terrain vertices. This error in the displacement of the radar cannot exceed $\frac{\sqrt{2}}{2}\Delta s$, where Δs is defined as the horizontal spacing of the vertices. The spacing between network vertices is 100 meters, so that the maximum displacement of the radar is 71 meters. The maximum possible error will therefore be $(P_{d|i}(R-71)/P_{d|i}(R)-1)\times 100\%$. This value is indistinguishable from zero for our data for R beyond 1600 meters (one mile): For R small, both of the probabilities are essentially one, the ratio is one and the error zero; when R is sufficiently large that the probabilities are less than 0.99 (about 20 miles), 71 meters is a such a small fraction of the range, the probabities are nearly identical, the ratio is nearly one and the error is zero.

b. Error from Approximating the Distance from the Radar

In order to obtain a point estimate for the returned signal strength an estimate of the range from the aircraft to the radar is necessary. There are several options available, three of which would be equally effective and have been considered seriously:

- use the maximum range from the radar on the edge,

- use the minimum range or Closest Point of Approach (CPA) from the radar on the edge, or
- use the average range from the radar on the edge.

Using the maximum range from the radar will minimize the returned signal strength, providing a lower bound of sorts for the probability of detection and subsequently an overly optimistic view of the route. Likewise, using the minimum range from the radar would provide an upper bound of sorts. While using the minimum range would be pessimistic from a planning perspective, the average range allows for a more accurate calculation without being overly optimistic.

This model uses the average range to the radar, as it more closely represents the range of the aircraft over the entire edge. This approximation has the least impact on the calculation of returned signal strength when the edge is tangent to the arc of the radar scan; it is also equivalent to CPA when the edge is tangent. This approximation has the most significant impact when the edge is heading directly at the sensor.

The error in the approach taken can certainly be bounded by the difference in values of $P_{d_{i}}(R)$ when R takes on minimum and maximum values along a network edge. Assume that the radar is located on a vertex and that the aircraft is moving on a network edge directed straight toward the radar. If the aircraft is at an actual distance of R meters, the error in probability of detection will be no worse than if the range of the vertex at the other end of the edge at $R-100$ meters is used in computations.

The resulting error $(P_{d|i}(R-100)/P_{d|i}(R)-1)\times 100\%$ is nearly zero for all values of R beyond 1600 meters, as before.

c. *Error from Approximating Exposure Time*

Since probability of detection is dependent on exposure time, and current detection minimization techniques embrace terrain masking to hide from a sensor, it is important to consider the case that an aircraft may only be exposed to a radar for a portion of the edge on which it is flying. To facilitate calculation when building a test program, it is assumed that for physically short edges we could approximate the exposure time as either full exposure over the edge, exposure on half of the edge, or completely masked over the full edge. We will use techniques similar to those employed in the range approximation error analysis to explore the errors incurred by this approximation.

Approximating the exposure time affects the probability of detection by changing the value for $c(t)$, the number of scans of the aircraft in time t. Using pessimistic values for $P_{d|i}(R)$, and for $P_i(t)$, and considering the actual exposure time to be qt, a percent change of the estimate, it is a simple matter to find the relative error.

Recall that $P_d(t,R) = 1 - [1 - P_{d|i}(R)]^{c(t)}[1 - P_i\left(t - \frac{c(t)w}{6s}\right)P_{d|i}(R)]$ and that $c(t) = \left\lfloor \frac{6qts}{360 - w} \right\rfloor$. By replacing t with qt, $P_{d|i}(R)$ with 0.5, and $P_i(t)$ with 1.0, a pessimistic $P_d(qt,R)$ can be found: $P_d(qt,R) = 1 - [0.5]^{\frac{6qts}{360 - w} + 1}$. By dividing the original $P_d(t,R)$ by this quantity, the relative error can be found to be $(0.5)^{q-1}\%$. The largest error possible would occur when q=0, or when the aircraft was not exposed to the radar, but full exposure is assumed. In this case the percentage error is $(0.5)^{0-1}$=2%.

V. RECOMMENDATIONS FOR FUTURE RESEARCH

A viable method for selecting a good flight path between two points has been established. Both the basic algorithm and its underpinning models have been validated through a battery of simulated missions. This framework must now be built upon to provide a useable product to the fleet. The following paragraphs detail some specific recommendations in this regard.

A. MODEL IMPROVEMENT

In the development of this model, several assumptions were made and certain inefficiencies were left that may affect the quality of the route. There may be ways to either eliminate the assumptions or inefficiencies or lessen their effects.

1. Independence of Detections

In order to use the probability of detection equations developed in Chapter II, it was necessary to assume that detection on coincident edges in a flight path are independent. Given the relatively short physical edge length this is not likely to be true, as discussed in Chapter II. Because of this disparity between assumed conditions and reality, the absolute quality of the selected route cannot be determined, and should not be used in any decision-making process as to the viability of a mission. If absolute route quality is desired, it is necessary to find some way to lessen the impact of this assumption.

One way to accomplish this might be to label a vertex when it is encountered, noting to which radars it is exposed. Then, when calculating the exposure time along the route, the total time that the aircraft is exposed to a set of radars can be used. Then, each time this exposure factor changes along the route, or if the route changes direction, the route may be broken into longer legs, assuming independence between them. This adjustment would have the effect of allowing straight portions of the flight path to grow in length until the radar coverage changes. The route could still be selected based on probability of detection. This does not eliminate the independence assumption, but it might serve to reduce the error in calculations. Also note that other problems arise when using longer or variable length legs, such as the approximation of the range to a particular radar when calculating the probability of detection. This problem demands more research and if a satisfactory solution can be found it should be incorporated into the model.

2. Network Model Resolution

For flat terrain or regions with a low concentration of radar sensors, the high resolution of the model makes the prototypic algorithm very inefficient. However, it may be that a given operating area may have mixed terrain. One example of this might be any water access over a beach to foothills over mountains to an objective on the other side, similar to aircraft carrier based training missions to the California and Nevada deserts. We recommend that a terrain recognition algorithm be devised that can be implemented to preprocess the input data. This algorithm should be able to recognize relatively gentle terrain and reduce the resolution accordingly. When it encounters more rugged terrain it

should increase the resolution of the network. Recognition might be accomplished by measuring the angle between terrain elevation observations in the DTED file. Having differing resolutions in different regions of the network should not adversely affect the results, but it should improve efficiency. Time would still be calculated on each edge using speed and distance, and everything else in the model would remain the same. For each vertex eliminated an additional 24 nearest neighbors can be eliminated in the search. This will greatly enhance the search efficiency, could reduce the impact of the independence assumption by physically lengthening the edges between vertices, and may improve accuracy.

3. Route Restrictions

The current prototype chooses a flight path without regard to distance limitations. Realistically, an aircraft has a fixed maximum range, known as its "combat radius," which is directly linked to its fuel capacity (*NATOPS Flight Manual*, 1991). For a helicopter in cruise flight, a common assumption in route planning is a constant fuel burn rate. This assumption allows for a reasonably fixed distance that can be used for flight planning. If the number of vertices that must be considered by a search algorithm can be reduced by use of this information it would greatly enhance search efficiency, reduce memory requirements, and help ensure that the route selected is not too long.

The locus of points equidistant from two fixed points is an ellipsoid. That is, if one took any set of points on an ellipsoid and compared the combined distances from that ellipsoid's foci, then those combined distances would be equal. If a solid ellipsoid was formed so it was completely inscribed in the currently box-shaped network, it would

contain roughly half of the vertices in the entire network. Eliminating half of the vertices will have a significant effect on the efficiency of the search algorithm. We propose that the combat radius be used as the combined distance from the start and goal vertices to form a solid ellipsoid of vertices that may comprise a viable route through the network. Using the combat radius may allow us to create a smaller ellipsoid, further reducing the number of vertices required to be searched and providing even greater gains in efficiency. This technique would not ensure that the selected route would be shorter than the maximum allowed. In order to accomplish that goal, additional constraints on the problem must be included. We recommend that the maximum range ellipsoid be used to improve the efficiency of the algorithm. This can be done within the search algorithm by ignoring vertices outside the ellipsoid.

Additional constraints on route selection might be developed to ensure that the route selected is practical. "The constrained shortest-path problem" is well-known to be NP-complete, but a pseudo-polynomial time algorithm could be developed that expands the network vertices by distance, or time, or fuel consumption, etc. Such an algorithm could be very inefficient and before implementing such an algorithm, a "lagrangian relaxation approach," e.g., (Ahuja, et al., 1993, pp. 598-602), should probably be tried. Using this approach, to constrain distance say, would add a positive multiplier times the physical length of an edge to the edge's current "length" which is a "probability weight". In this way, a path would be penalized for its length. By adjusting the multiplier, it may be possible to obtain a path that, at least approximately, minimizes probability of detection while being no longer than a specified length.

These modifications are not minor in nature, but could have a significant effect on further development of this model.

4. Improvements in Algorithmic Complexity

As discussed in Chapter III, the nearest neighbors of a vertex, and hence the next set of vertices to be considered for the route during the algorithm, can be calculated rather easily. Currently, the algorithm examines all of the nearest neighbors of a vertex when it is encountered, except the previous vertex. The algorithm could be altered to further reduce the set of vertices to be considered for examination. One suggestion is to eliminate all of the vertices that represent a reversing of course for the aircraft; that is, any vertices that lie behind a plane perpendicular to the most recent edge transited may not be considered for addition to the route. By making such a restriction, we might be able to improve the efficiency of the algorithm by up to 50%. This type of restriction might require some additional, complicated constructs in the algorithm, however, so the topic needs to be studied further.

5. Output

We recommend that an output algorithm be developed to provide a more useable flight path for the flight crew. Current output is in terms of network edges and has waypoints approximately 100-150 meters apart. We suggest that the raw flight path be smoothed, and waypoints, heading, and altitude information be extracted and put into a format that is readily useable by the aircrews.

Since the navy uses TAMPS to evaluate the quality of a flight path that is selected using current methods, the output from this algorithm might be processed so that it may be directly input into TAMPS. In this manner, some form of absolute route quality may be determined, and the graphics and route analysis capability of TAMPS may be fully utilized, while saving valuable route planning time.

6. Bugs

Errors encountered in solving problems with nonzero detection probabilities point to difficulties with the series approximations used to compute detection probabilities. These numerical difficulties with published methods need to be investigated and corrected for the methodology of this thesis to be viable.

LIST OF REFERENCES

Aho, Alfred V., Hopcroft, John E., Ullman, Jeffrey D., *Data Structures and Algorithms,* Reading, Pennsylvania, Addison-Wesley Publishing Company, 1983.

Ahuja, Ravindra J., Magnanti, Thomas L., and Orlin, James B. *Network Flows,* Englewood Cliffs, New Jersey: Prentice Hall, 1993.

Ammann, Joann M., McGhee, Robert B., and Zyda, Michael J., "User Interface Design for Two-Dimensional Polygonally Encoded Geological Survey Maps," Naval Postgraduate School, Monterey, CA, Report Number NPS52-86-017, July 1986.

Bailey, Michael P., "Measuring Performance of Integrated Air Defense Networks Using Stochastic Networks," *Operations Research,* Vol. 40, 1992, pp. 647-659.

Bailey, Michael P., Crane, Allen D., and Miluski, J. Peter, "The Prowler IADS Performance Evaluation Tool (PIPE)," Naval Postgraduate School, Monterey, CA, Report Number NPS-OR-94-009, May, 1994.

Diehl, Robert K., McGhee, Robert B., and Zyda, Michael J., "Two-Dimensional Polygonal representations of Maps for Use with Autonomous Vehicle Route Planning," Naval Postgraduate School, Monterey, CA, Report Number NPS52-86-016, June 1986.

Hartman, James K., "Parametric Terrain and Line of Sight Modelling in the STAR Combat Model," Master's Thesis, Naval Postgraduate School, Monterey, CA, August 1979.

McGhee, Robert B., Zyda, Michael J., Smith, Douglas B., Streyle, Dale G., "An Inexpensive Real-Time Interactive Flight Simulation System," Naval Postgraduate School, Monterey, CA, Report Number NPS52-87-034, July 1987.

Microsoft FORTRAN Reference, Version 5.1, Redmond, WA, Microsoft Corporation, 1991.

Mitchell, Joseph S. B., "Planning Shortest Paths," Ph.D.Dissertation, Stanford University, Stanford, CA, 1986.

NATOPS Flight Manual, Navy Model SH-3G Aircraft, NAVAIR 01230HLC-1, January 15, 1991.

Ong, Seow Meng, "A Mission Planning Expert System with Three-Dimensional Path Optimization for the NPS Model 2 Autonomous Underwater Vehicle," Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1990.

Operational Test and Evaluation Force, "Follow-on Operational Test and Evaluation of the Tactical Aircraft Mission Planning System," August 1991.

Richbourg, Robert F., "Solving a Class of Spatial Reasoning Problems: Minimal-Cost Path Planning in the Cartesian Plane," Ph. D. Dissertation, Naval Postgraduate School, Monterey, CA, June 1987.

Skolnik, Merrill I., *Radar Handbook*, New York, McGraw-Hill Publishing Company, 1980.

Systems Research Group, *Annual Progress Report: The Tank Weapon System*, Columbus, OH, Department of Industrial Engineering, Ohio State University, June 1966.

Washburn, Alan R., *Search and Detection*, Arlington, Virginia, ORSA Books, 1989.

Wrenn, Lawrence R III., "Three-Dimensional Route Planning for a Cruise Missile for Minimal Detection by Observer," Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1989.

Zyda, Michael J., McGhee, Robert B., and Taylor, Gary W., "Parametric Representation and Polygonal Decomposition of Curved Surfaces," Naval Postgraduate School, Monterey, CA, Report Number NPS52-86-028, December 1986.

INITIAL DISTRIBUTION LIST

No. copies

1. Defense Technical Information Center — 2
   Cameron Station
   Alexandria, VA 22304-6145

2. Library, Code 52 — 2
   Naval Postgraduate School
   Monterey, CA 93943-5101

3. Professor R. Kevin Wood — 8
   Department of Operations Research
   Naval Postgraduate School, Code OR/Wd
   Monterey, CA 93943-5008

4. Professor Craig W. Rasmussen — 2
   Department of Mathematics
   Naval Postgraduate School, Code MA/Ra
   Monterey, CA 93943-5008

5. Chief of Naval Operations (N-81) — 1
   Navy Department
   Washington, DC 20350

6. Mr John J. Leary III — 2
   c/o Professor R. Kevin Wood
   Department of Operations Research
   Naval Postgraduate School, Code OR/Wd
   Monterey, CA 93943-5008